# Comparison of clustering techniques for hybrid rocket fuel combustion data

International Workshop on Scientific Machine Learning
Cologne, January 8-10, 2020

Alexander Rüttgers*, Charlotte Debus*, Martin Siggel*
Anna Petrarolo**, Mario Kobald**

*Institute for Simulation and Software Technology
** Institute for Space Propulsion
German Aerospace Center (DLR)

# Motivation (ATEK research rocket flight in June 2019)

https://www.youtube.com/watch?v=JlcReUwZXFU



# Outline

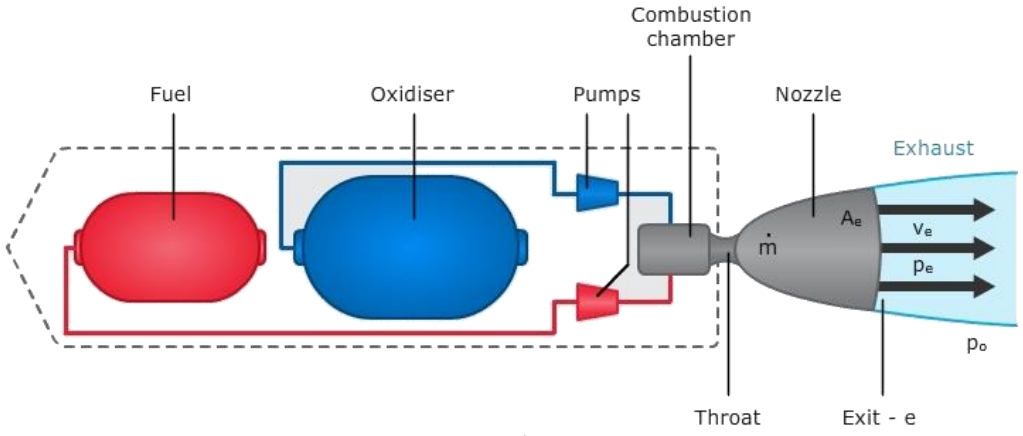
1. Rocket engine combustion analysis at DLR
2. Helmholtz Analytics Toolkit (HeAT) for distributed ML
3. Clustering results with HeAT

# Outline

1. Rocket engine combustion analysis at DLR
2. Helmholtz Analytics Toolkit (HeAT) for distributed ML
3. Clustering results with HeAT

# Rocket engine combustion analysis

- **Aim:** Cost reduction of rocket engines, be competitive with e.g. Space-X

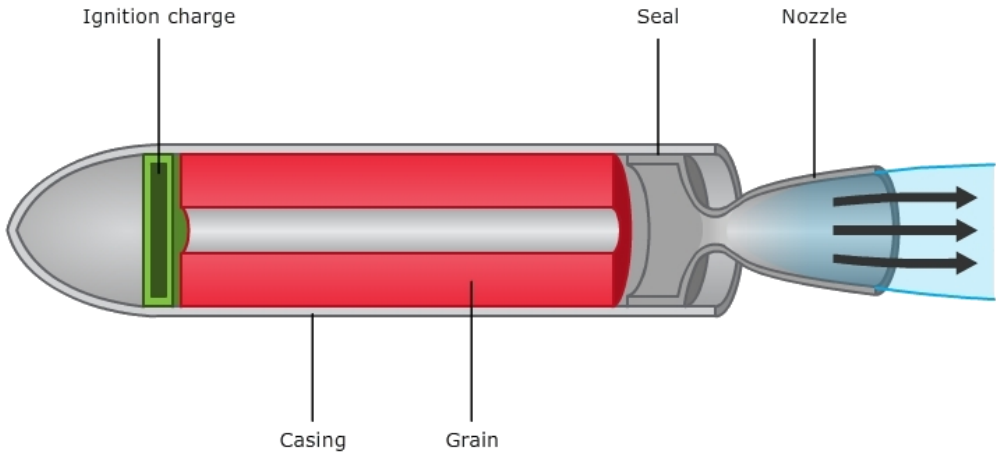



## Traditional liquid rocket engine:

- 2 pumps transporting fluid fuel and oxidizer at very high pressure and flow
- Advantages
  - Burning rate can be controlled precisely
- Disadvantages
  - Pumps are mechanically very complex
  - Expensive

# Rocket engine combustion analysis

- **Aim:** Cost reduction of rocket engines, be competitive with e.g. Space-X

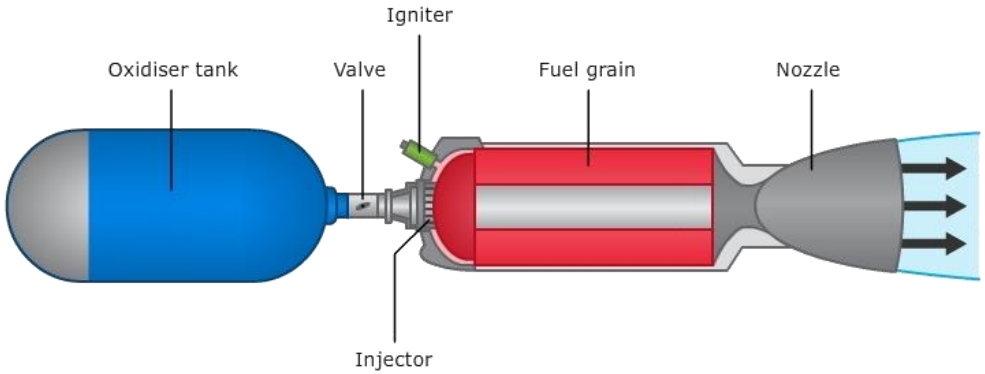



## Solid propellant rocket engine

- Fuel and oxidizer are mixed in solid form
- Advantage
  - Cheap
- Disadvantage
  - Burning rate can not be varied during flight

# Rocket engine combustion analysis

- **Aim:** Cost reduction of rocket engines, be competitive with e.g. Space-X




## Hybrid rocket engine

- Pressurized fluid oxidizer
- Solid fuel
- A valve controls, how much oxidizer gets into the combustion chamber

- Advantages
  - Cheap
  - Controllable

# Project ATEK: Experiments on new hybrid rocket fuels at DLR

- DLR investigates new hybrid rocket fuels on a paraffin basis at Institute of Space Propulsion in Lampoldshausen.

- About 300 combustion tests were performed with single-slab paraffin-based fuel with 20° forward facing ramp angle + gaseous oxygen.

- Two different fuel compositions:
  - pure paraffin 6805
  - paraffin 6805 + 5% polymer

**Fig. 1:** Fuel slap configuration before (top) and after (bottom) combustion test.

# Combustion chamber set-up

- Optically accessible combustion chamber is 450 mm long, 150 mm wide and 90 mm high.
- Tests were performed with different configurations (e.g. fuel, oxidizer mass flow, filters)
- Combustion is captured with high-speed video camera with 10 000 frames / second

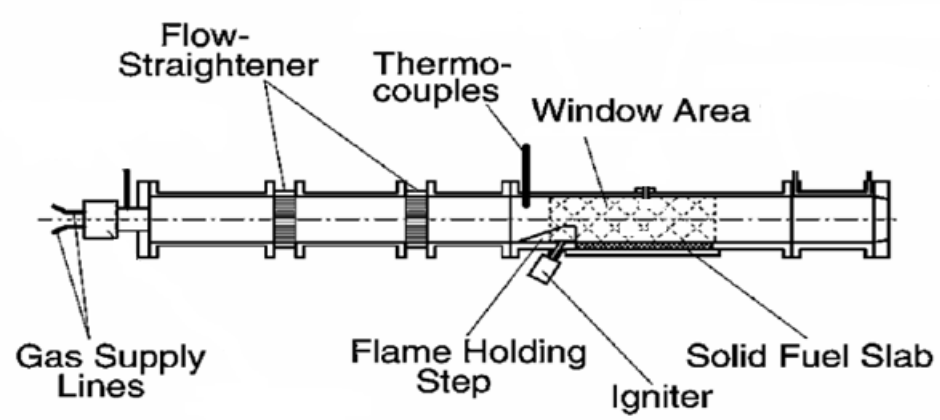
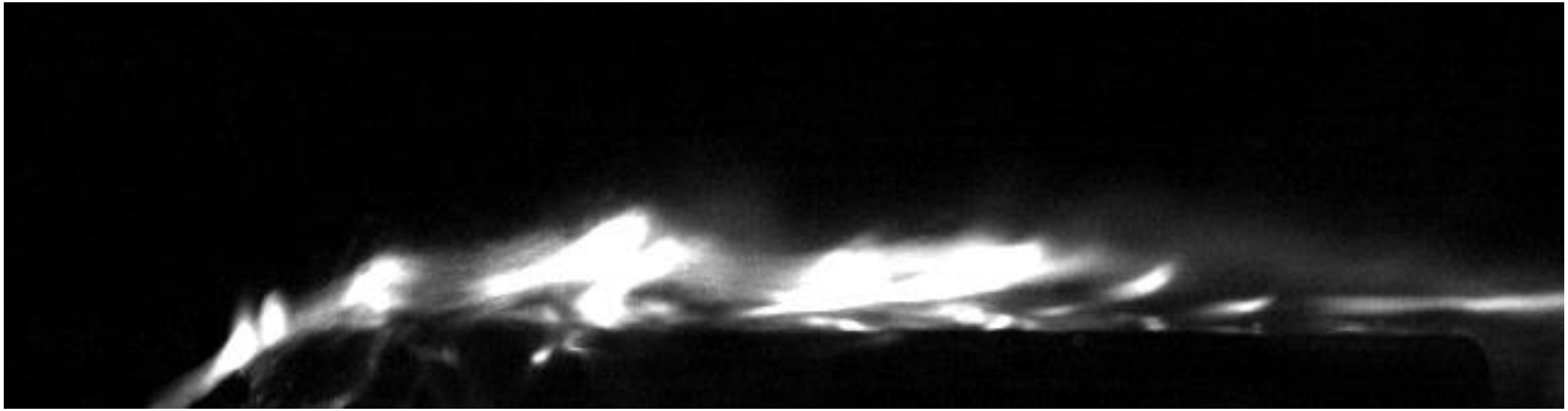



**Fig. 2:** Side view of combustion chamber

| Test no. | Fuel | | $\dot{m}_{Ox}[g/s]$ | | CH* filter |
|---|---|---|---|---|---|
| | 6805 | 6805+5% polymer | 10 | 50 | |
| 284 | ✓ | | | ✓ | ✓ |
| 289 | | ✓ | | ✓ | ✓ |
| 296 | | ✓ | ✓ | | ✓ |
| 243 | | ✓ | ✓ | | |

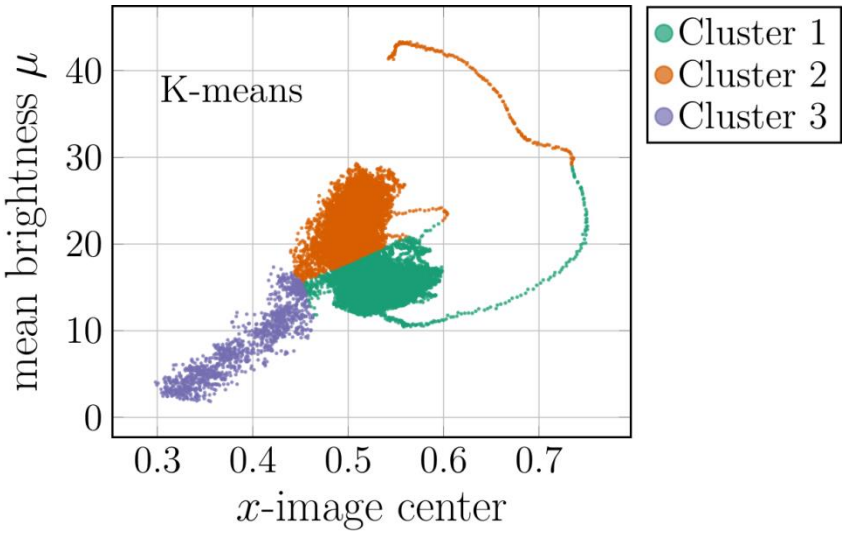
**Fig. 3:** Test matrix used for data analysis



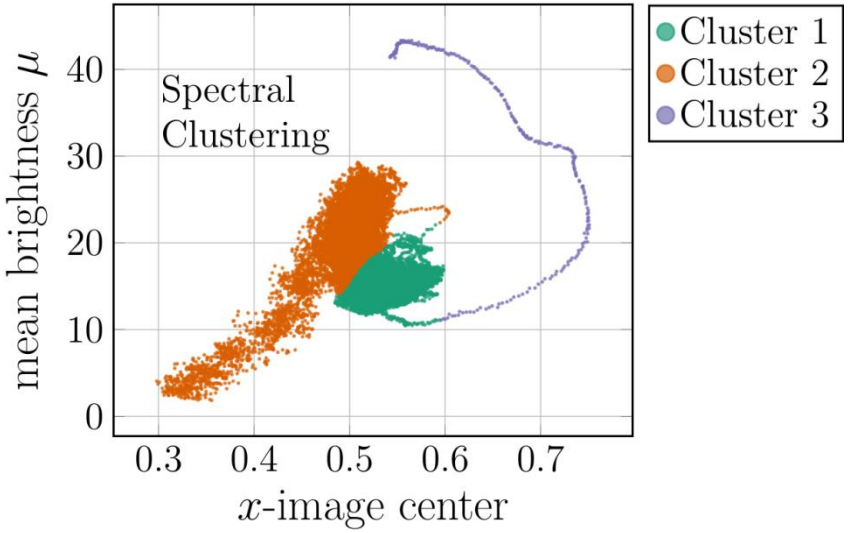
| Video extract of test 284 | fuel | oxidizer mass flow | CH*-filter | duration |
|---|---|---|---|---|
| Ignition, steady combustion, extinction | pure paraffin 6805 | 50 *g/s*, | yes, only wavelengths emitted from CH* are filmed | 3 s = 30 000 frames / 8GB raw data per test |

# Clustering of combustion image data

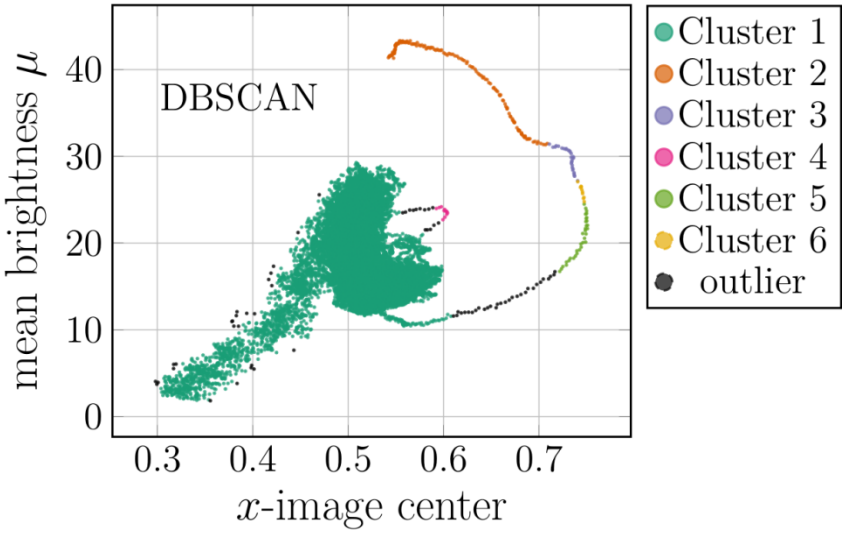
- Clustering of combustion data = identify different phases of the flow.
- Various clustering algorithms exist in the literature (DBSCAN, spectral clustering, k-means, …).
- **Start:** Comparison of algorithms on two features $(\mu, \overline{x})_j$ for all $j = 1, \ldots, 30000$ images of test 284.




computing time $\approx 0.8\ s$




computing time $\approx 70\ s$



# Comparison of clustering algorithms for presented application

| | K-means | Spectral clustering | DBSCAN |
|---|---|---|---|
| approach | • Iteratively minimize the within-cluster sum of squares | • Construct similarity matrix $\boldsymbol{A}$ of size (nr_of_points)x(nr_of_points)<br>• Build graph Laplacian matrix $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$ with diagonal matrix $D_{ii} = \sum_j A_{ij}$<br>• Compute first K eigenvectors of $\boldsymbol{L}$<br>• Cluster low-dimensional data representation with e.g. K-means | • Find points in $\varepsilon$-environment of every point<br>• If environment contains enough $minPts$ points, a new cluster is started<br>• Otherwise, it‘s noise or belongs to other cluster. |
| pros* | • Scales to large data sets<br>• Guarantees convergence<br>• Very simple | • Reduces curse of dimensionality<br>• Does not make strong assumptions on clusters (e.g. spherical shape) | • Does not require number of clusters K |
| cons* | • Choosing K manually<br>• Local optimal solutions<br>• Curse of dimensionality<br>• Similar-size clustering | • Choosing K manually<br>• Expensive for large datasets<br>• Number of hyperparameters | • Two hyperparameters $\varepsilon$ and $minPts$ that are hard to find if dataset is continuous |

*incomplete list

# Comparison of clustering algorithms for presented application

| | K-means (start first) | Spectral clustering (second approach) | DBSCAN (here not adequate) |
|---|---|---|---|
| approach | • Iteratively minimize the within-cluster sum of squares | • Construct similarity matrix $\boldsymbol{A}$ of size (nr_of_points)x(nr_of_points)<br>• Build graph Laplacian matrix $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$ with diagonal matrix $D_{ii} = \sum_j A_{ij}$<br>• Compute first K eigenvectors of $\boldsymbol{L}$<br>• Cluster low-dimensional data representation with e.g. K-means | • Find points in $\varepsilon$-environment of every point<br>• If environment contains enough $minPts$ points, a new cluster is started<br>• Otherwise, it‘s noise or belongs to other cluster. |
| pros* | • Scales to large data sets<br>• Guarantees convergence<br>• Very simple | • Reduces curse of dimensionality<br>• Does not make strong assumptions on clusters (e.g. spherical shape) | • Does not require number of clusters K |
| cons* | • Choosing K manually<br>• Local optimal solutions<br>• Curse of dimensionality<br>• Similar-size clustering | • Choosing K manually<br>• Expensive for large datasets<br>• Number of hyperparameters | • Two hyperparameters $\varepsilon$ and $minPts$ that are hard to find if dataset is continuous |

*incomplete list

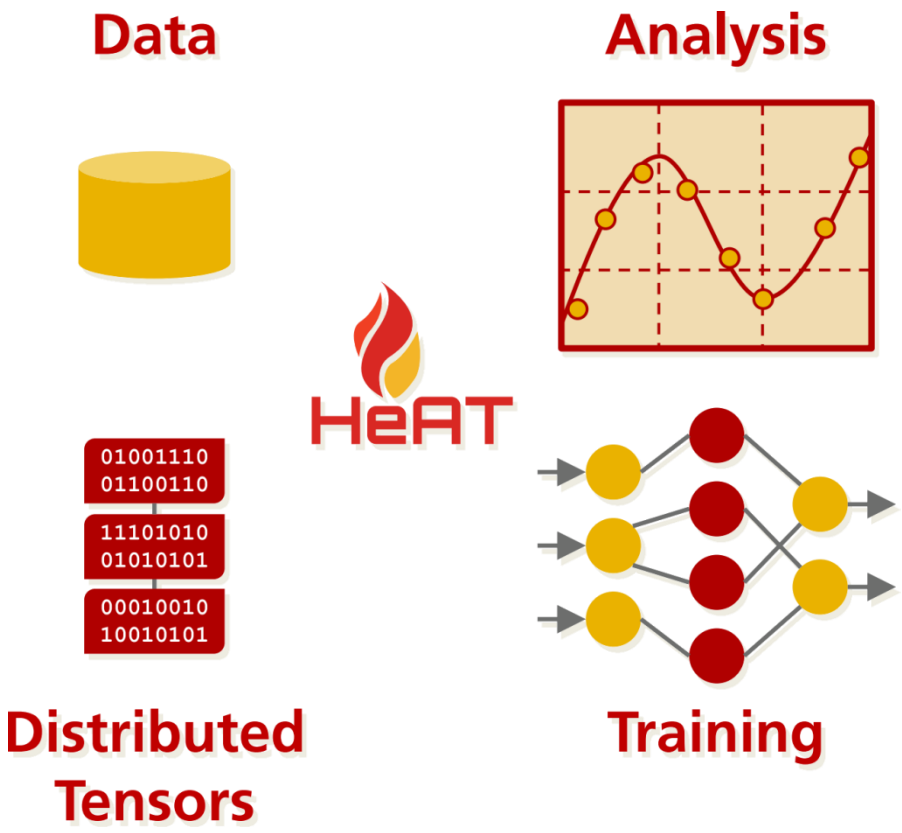
# Outline

1. Rocket engine combustion analysis at DLR
2. Helmholtz Analytics Toolkit (HeAT) for distributed ML
3. Clustering results with HeAT

# HeAT

- HeAT = Helmholtz Analytics Toolkit
- Python framework for **parallel**, **distributed** data analytics and machine learning
- Developed within the Helmholtz Analytics Framework Project since 2018
- **Aim:** Bridge data analytics and **high-performance computing**
- Open Source licensed, MIT

helmholtz-analytics/heat



# How we started HeAT:
## The Helmholtz Analytics Framework (HAF) Project

HELMHOLTZ
Analytics Framework

- Joint project of all 6 Helmholtz centers

DLR · JÜLICH Forschungszentrum · KIT Karlsruher Institut für Technologie · dkfz. DEUTSCHES KREBSFORSCHUNGSZENTRUM IN DER HELMHOLTZ-GEMEINSCHAFT · DESY · HelmholtzZentrum münchen German Research Center for Environmental Health

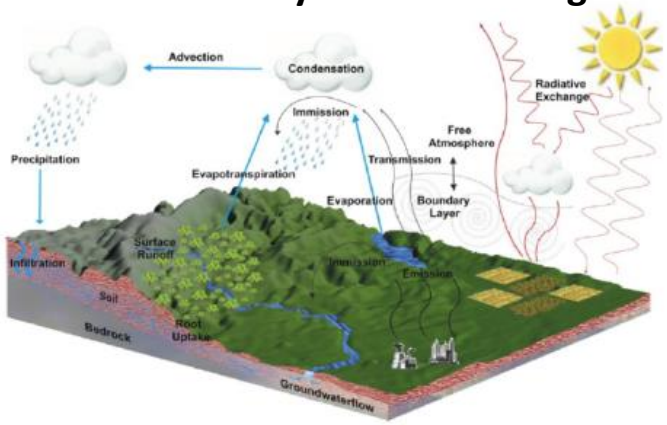
- Goal: foster data analytics methods and tools within Helmholtz federation.

- Scope:
  - Development of domain-specific data analysis techniques
  - Co-design between domain scientists and information experts

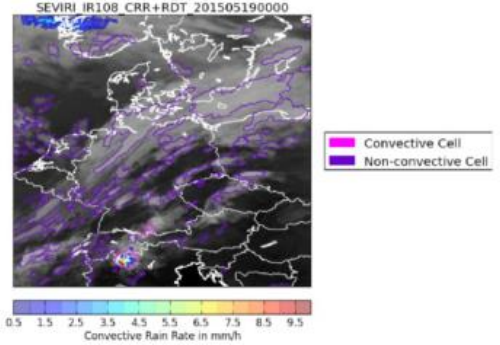


# Motivation: HAF applications

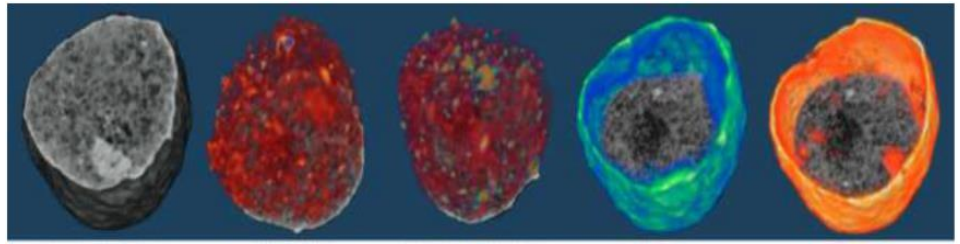
## Earth System Modelling



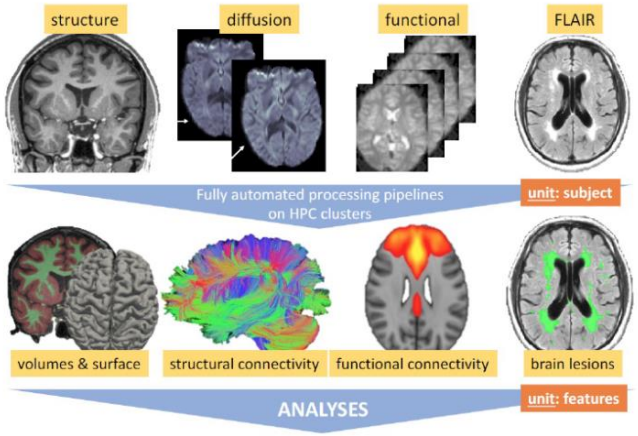

## SEVIRI Satellite Images – Near Real Time



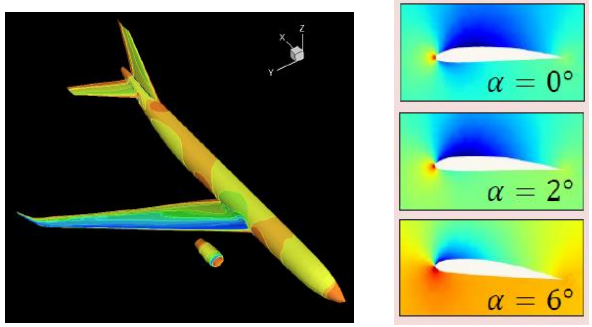

## Research with Photons



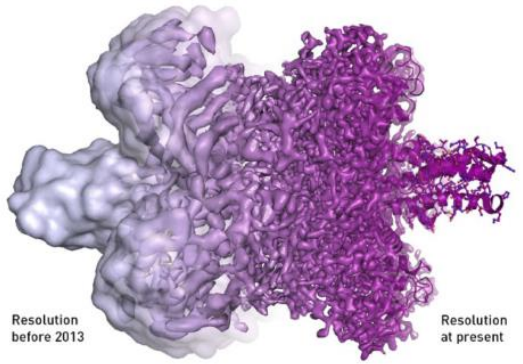
## Neuroscience




## Aeronautics and Aerodynamics




## Structural Biology



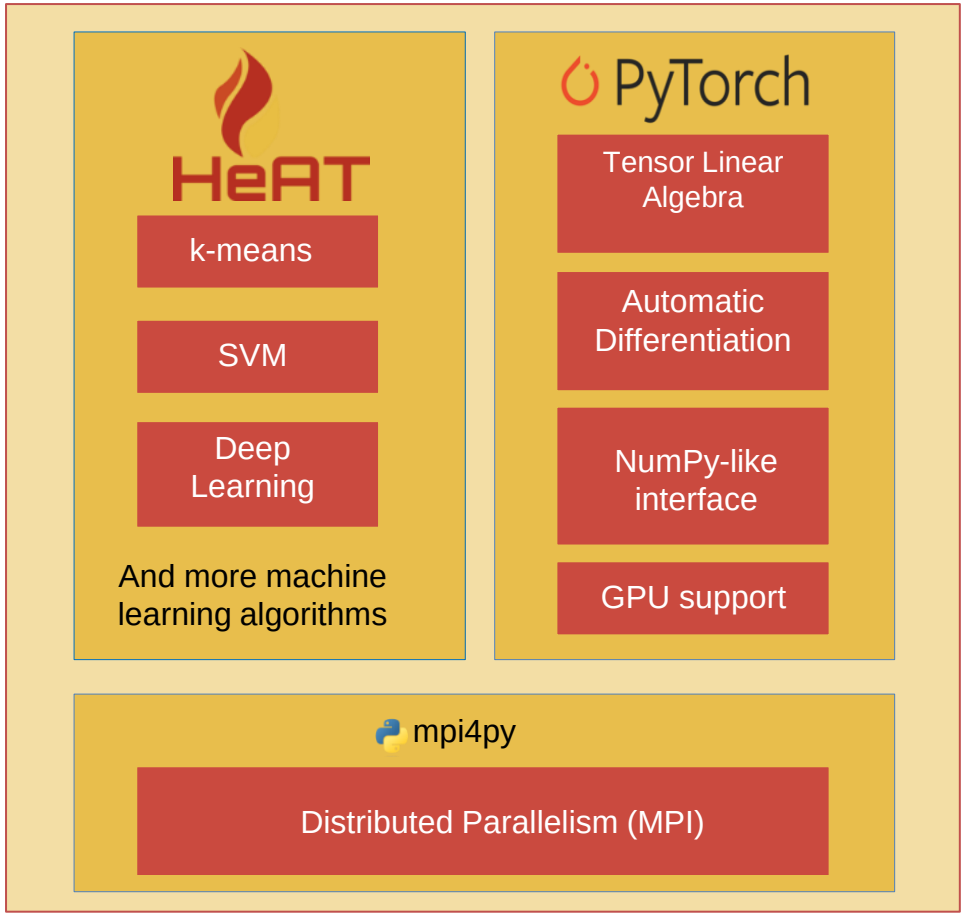
# Scope

Facilitating applications of HAF in their work

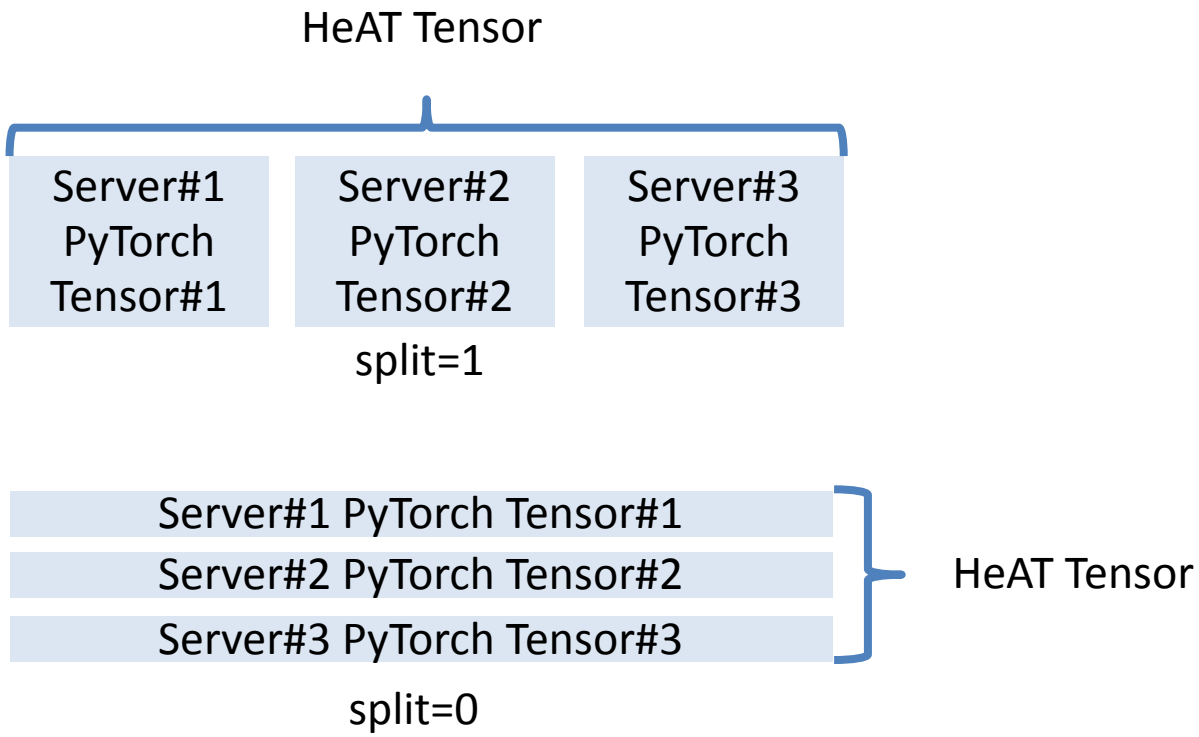
Bringing HPC and Machine Learning / Data Analytics closer together

Ease of use

# Design



# Data Distribution




Example:

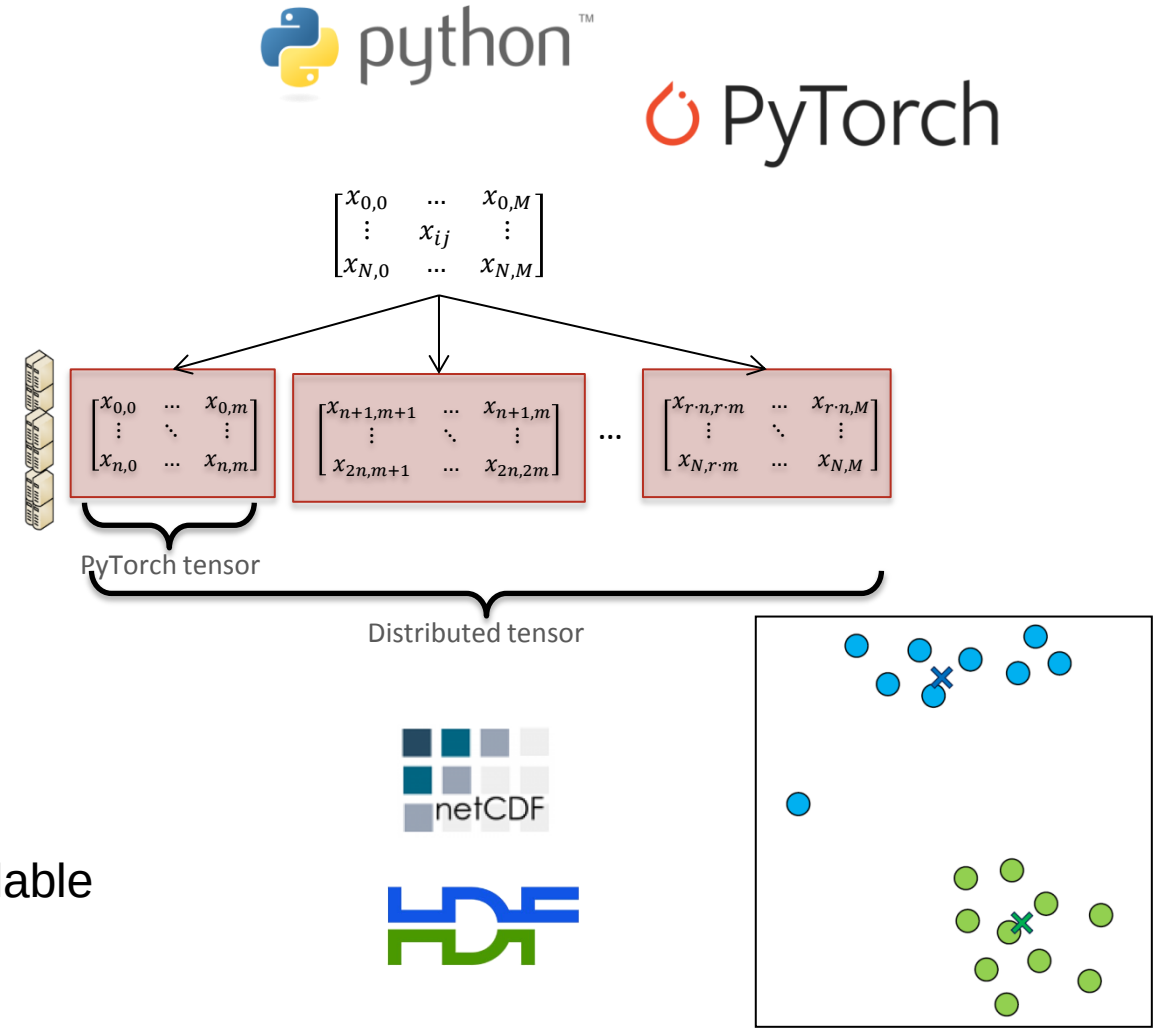
```
import heat as ht
# construct a range tensor
>>> range_data = ht.arange(6, split=1)
```

| Server#1 | Server#2 | Server#3 |
|---|---|---|
| [0, 1] | [2, 3] | [4, 5] |

```
>>> range_data.mean()
2.5
>>> range_data.argmax()
5
```

# What has been done so far?

- The core technology has been identified
- Implementation of a distributed parallel tensor core framework
- NumPy-compatible core functionality
- Some linear algebra routines
- Parallel data I/O via HDF 5 and NETCDF
- K-means and spectral clustering algorithms are available



# Outline

1. Rocket engine combustion analysis at DLR
2. Helmholtz Analytics Toolkit (HeAT) for distributed ML
3. Clustering results with HeAT

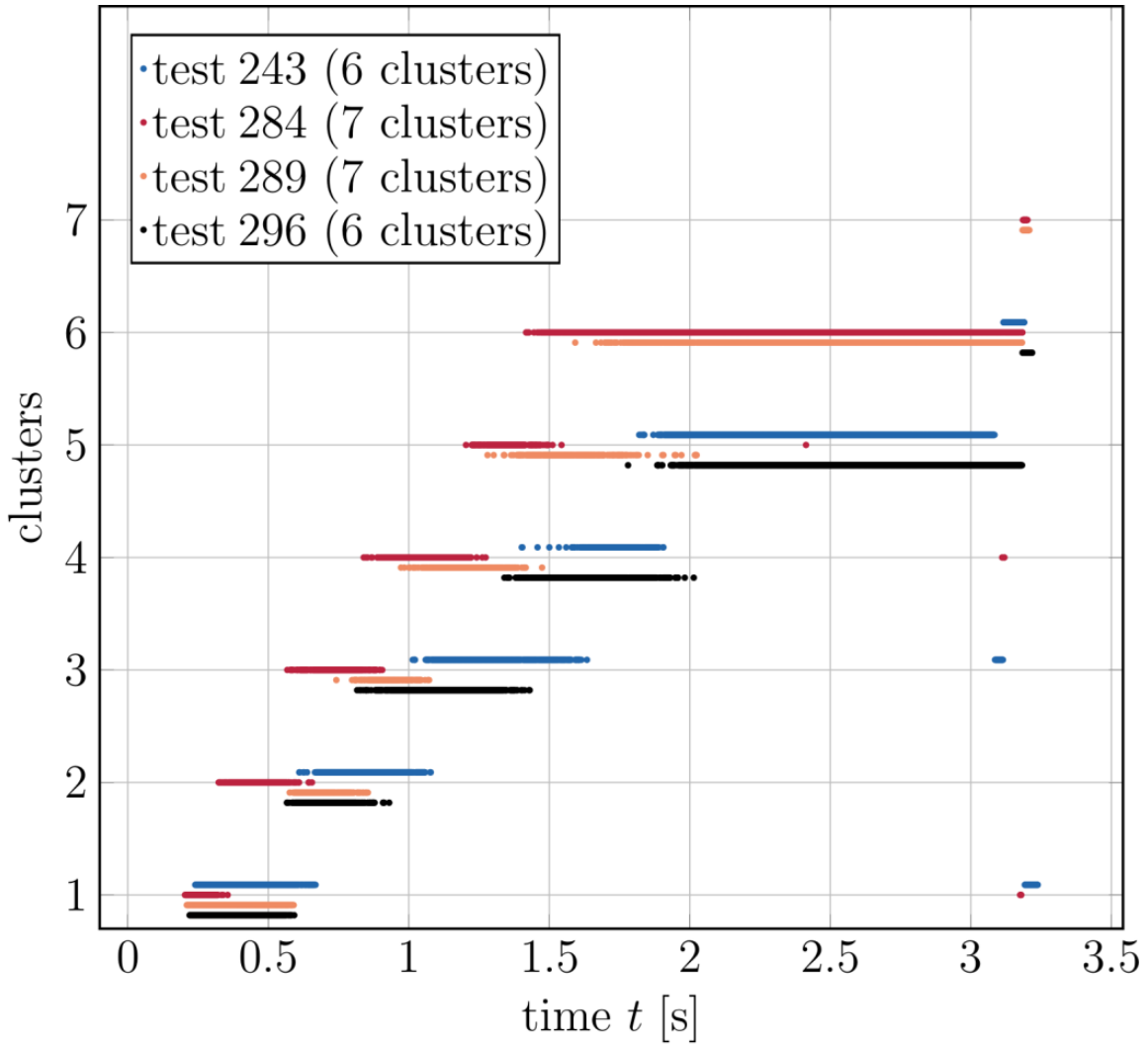
# K-means clustering: Strategies to avoid its drawbacks

- **Avoid local optimum solutions**
  - Algorithm is run multiple times (here: 10-times)
  - Take solution with smallest objective function (not a big difference in our case)
  - Implementation of K-Means++*
    - Choose the initial centers less randomly

- **Selection of K in K-means?**
  - Detailed analysis of objective function depending on *K* (here: algorithm is used for *K*= 2, …, 10)
  - Runtime of algorithm scales at least linearly in *K*
  - Note that an optimal *K* is often problem dependent

*Arthur and Vassilvitskii. K-means++: The advantages of careful seeding. SODA ’07, 2007





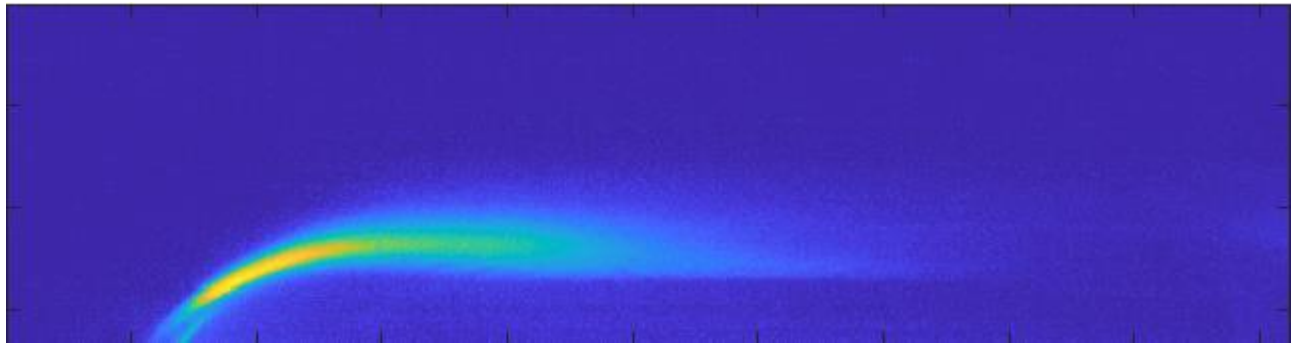

| Test | $C_1$ | $C_2$ | $C_3$ | $C_4$ | $C_5$ | $C_6$ | $C_7$ |
|---|---|---|---|---|---|---|---|
| 243 | 0.47 | 0.4 | 0.55 | 0.3 | **1.21** | 0.08 | x |
| 284 | 0.13 | 0.26 | 0.29 | 0.35 | 0.25 | **1.7** | 0.02 |
| 289 | 0.38 | 0.23 | 0.21 | 0.36 | 0.39 | **1.4** | 0.03 |
| 296 | 0.36 | 0.29 | 0.51 | 0.55 | **1.25** | 0.04 | x |

- Clustering allows for quantitative comparison.
- Apart from final cluster, all other clusters represent long-running flow phases.

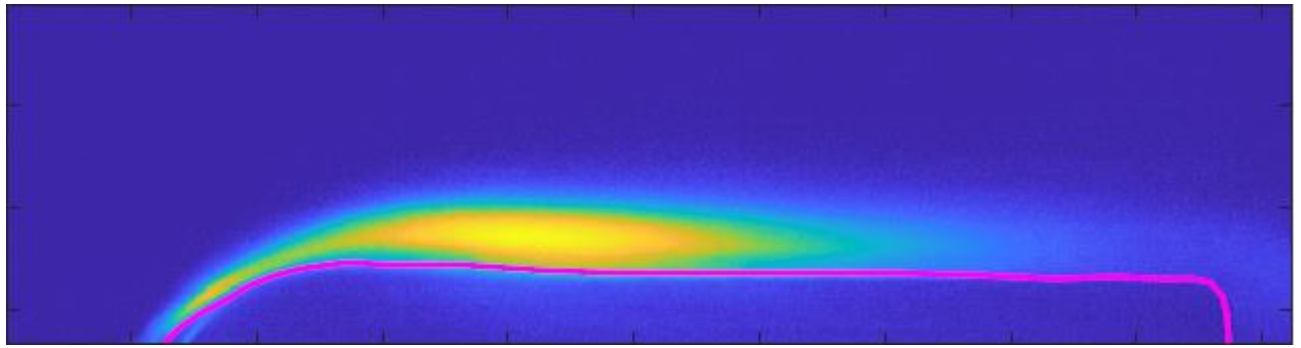
**Fig. 4:** Distribution of frames to their corresponding clusters.

**Fig. 5:** Time length of each cluster [s].

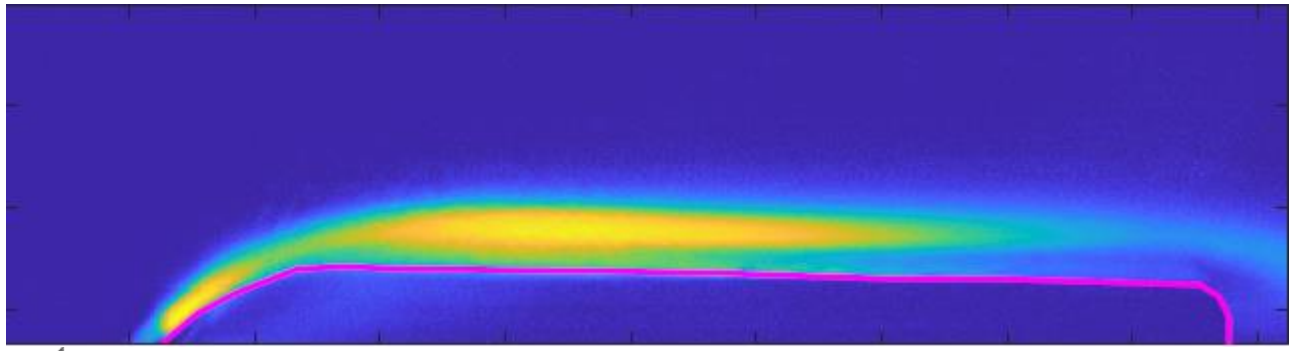
# Test 284 with K=7 (Part 1/3)



cluster 1
(1320 / 30000 frames)
ignition phase
(ignition comes from bottom of the chamber)

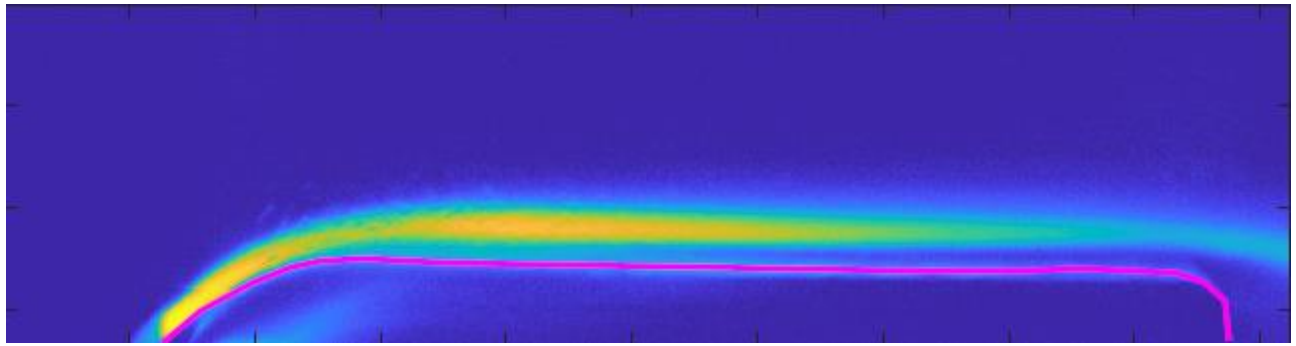


cluster 2
(2942 / 30000 frames)
burn phase without energy from outside
(ignition valves closed)

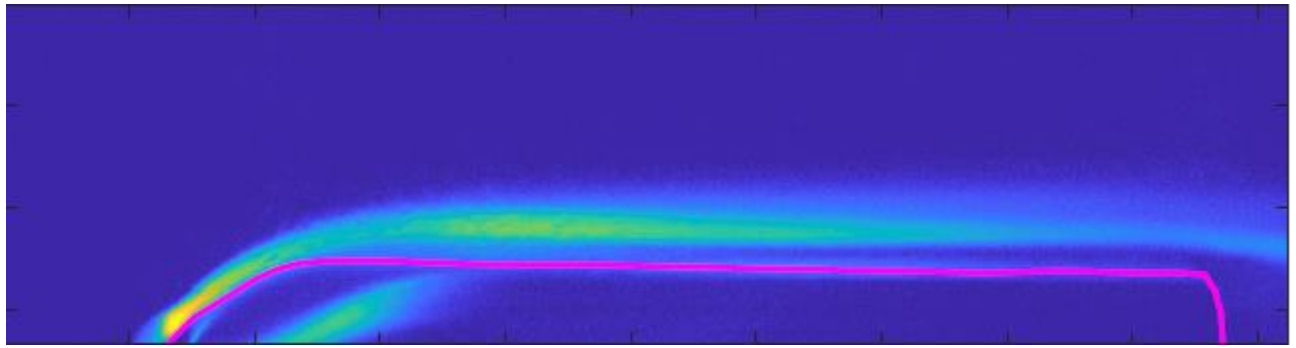


cluster 3
(3493 / 30000 frames)
fuel slap burns in the middle
(oxygen mass flow increases)

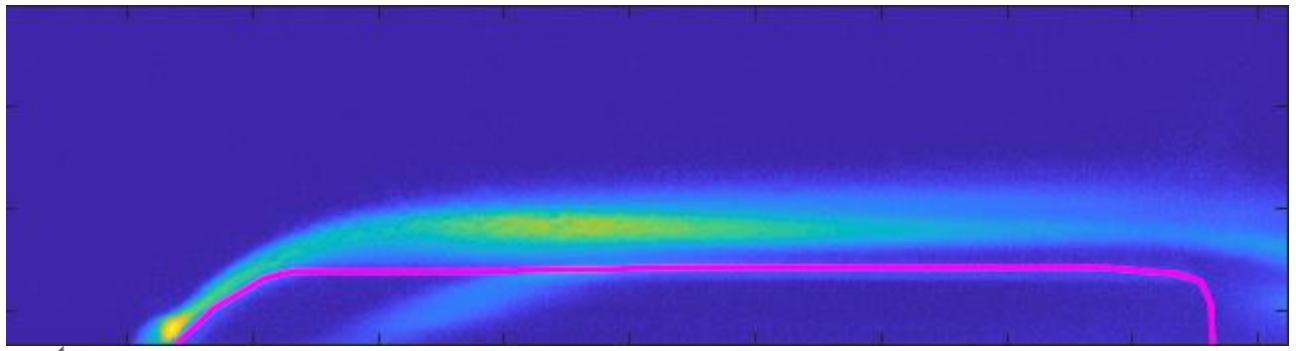
# Test 284 with K=7 (Part 2/3)



cluster 4
(3493 / 30000 frames)
whole surface is burning (brightness decreases due to CH*+O2 = CO+OH*)

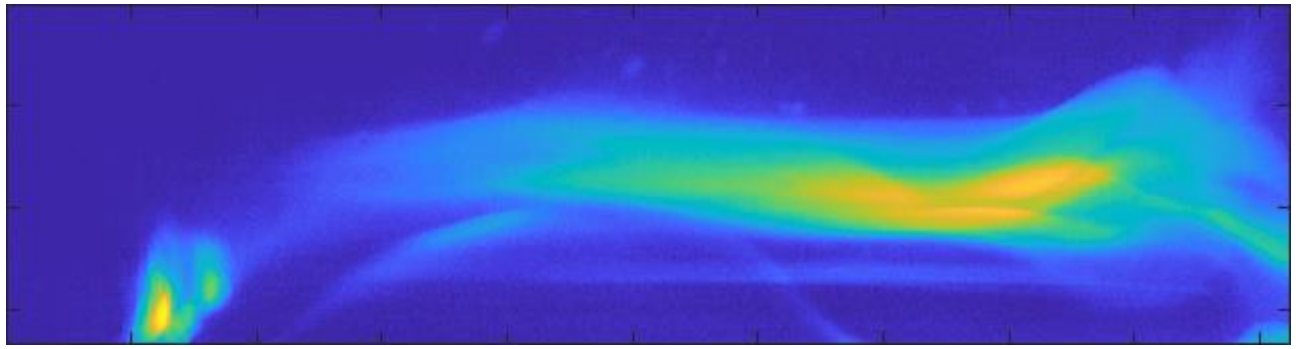


cluster 5
(2452 / 30000 frames)
large side flame close to camera



cluster 6
(16980 / 30000 frames)
constant combustion (with low CH* concentration, largest cluster in time)

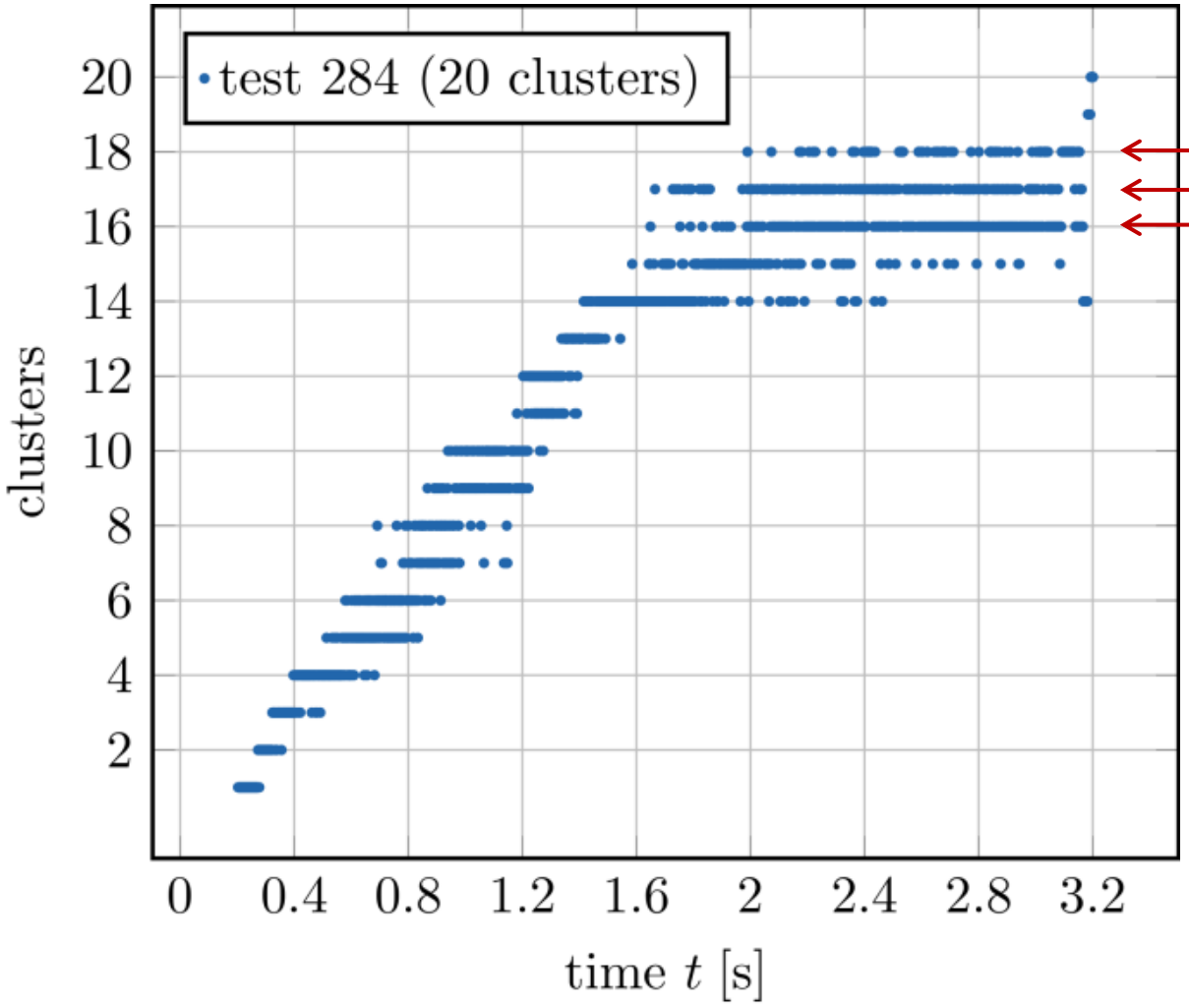
# Test 284 with K=7 (Part 3/3)



cluster 7
(194 / 30000 frames)
flame extinguishing phase (oxygen valve closes, nitrogen purge)

What about short-term irregularities?

# Increasing the number of clusters K?




overlapping clusters in time

**Solution strategies**

- cluster recombination / data postprocessing
- different clustering approach (e.g. spectral clustering)

# Spectral clustering: Strategies to avoid (some of) its drawbacks

- **Expensive for large datasets**
  - Usage of HeAT on HPDA-cluster at DLR
    - Distributed algorithm for similarity matrix computation
    - Implementation of distributed Lanczos algorithm for eigenvalue computation
  - Spectral clustering on 150 processes on 3 nodes took about 1 hour / test

- **Large number of hyperparameters?**
  - First results with HeAT have been achieved
  - Use scikit-learn + scikit-optimize / auto-sklearn / … on simplified problem (i.e. fewer images) to accelerate hyperparameter estimation



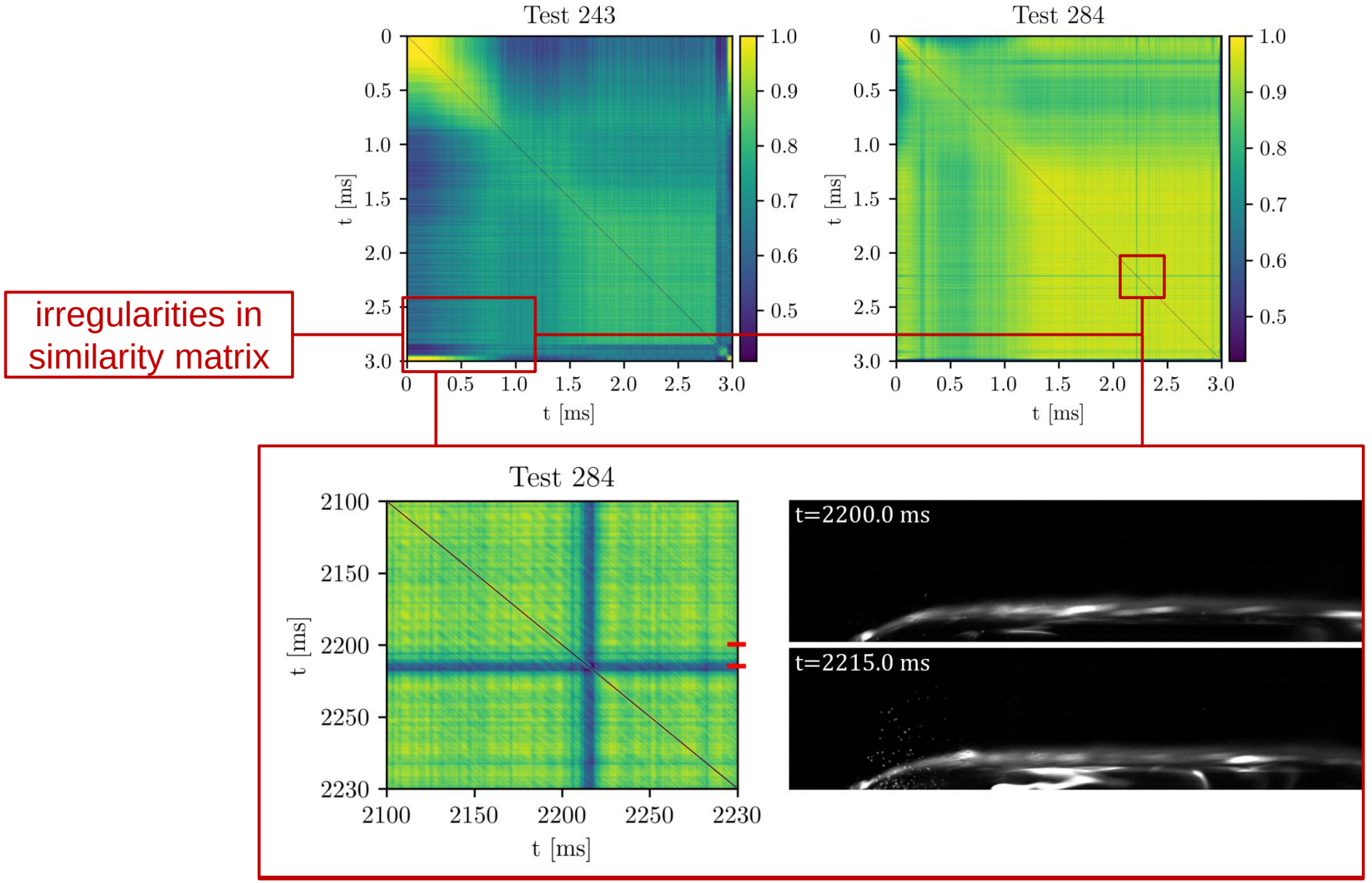



**Fig. 6:** Similarity matrix of all tests using a Gaussian kernel with variance $\sigma = 30000^2$.

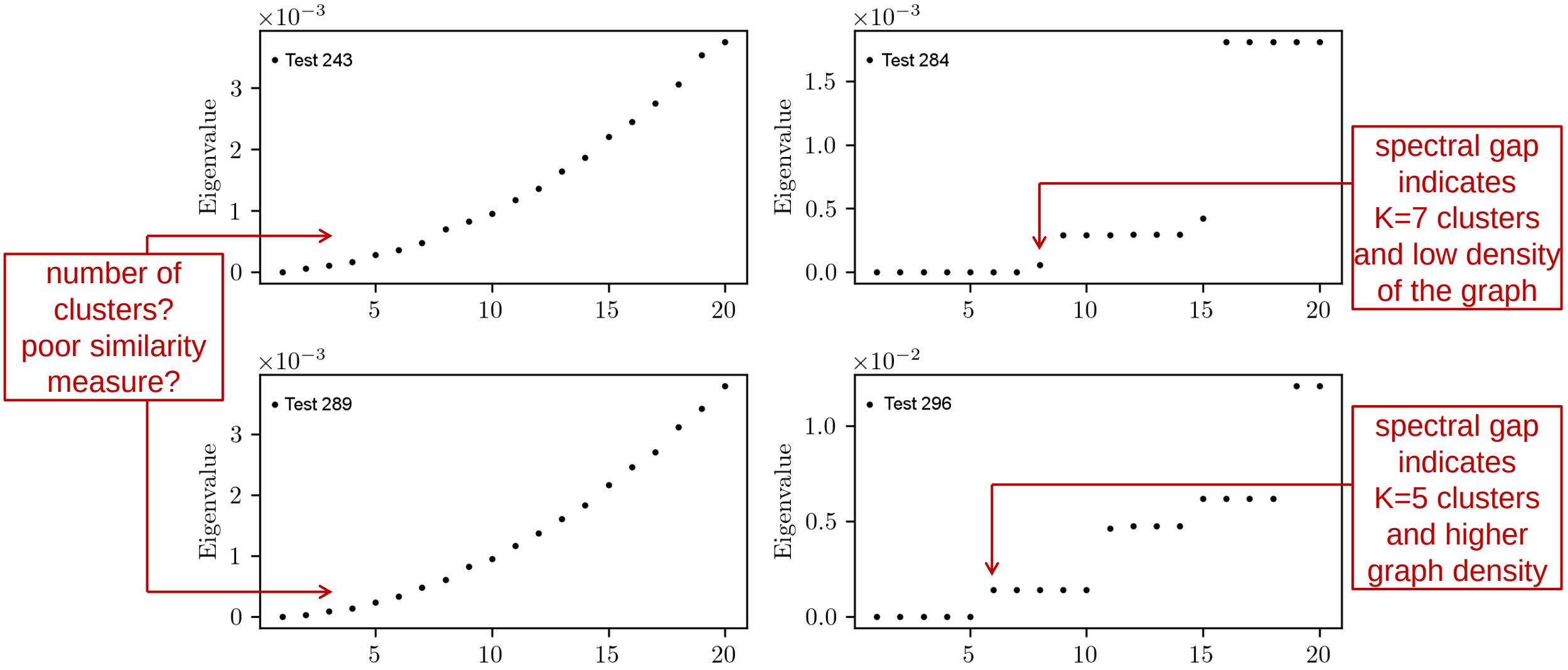



**Fig. 7:** 20 smallest eigenvalues of the graph Laplacian of all four tests. The number of 0 eigenvalues of the graph Laplacian corresponds to number of connected components.*

*Von Luxburg. Statistics & Computing, 17:4 (2007)

# Hyperparameter optimization of test 284

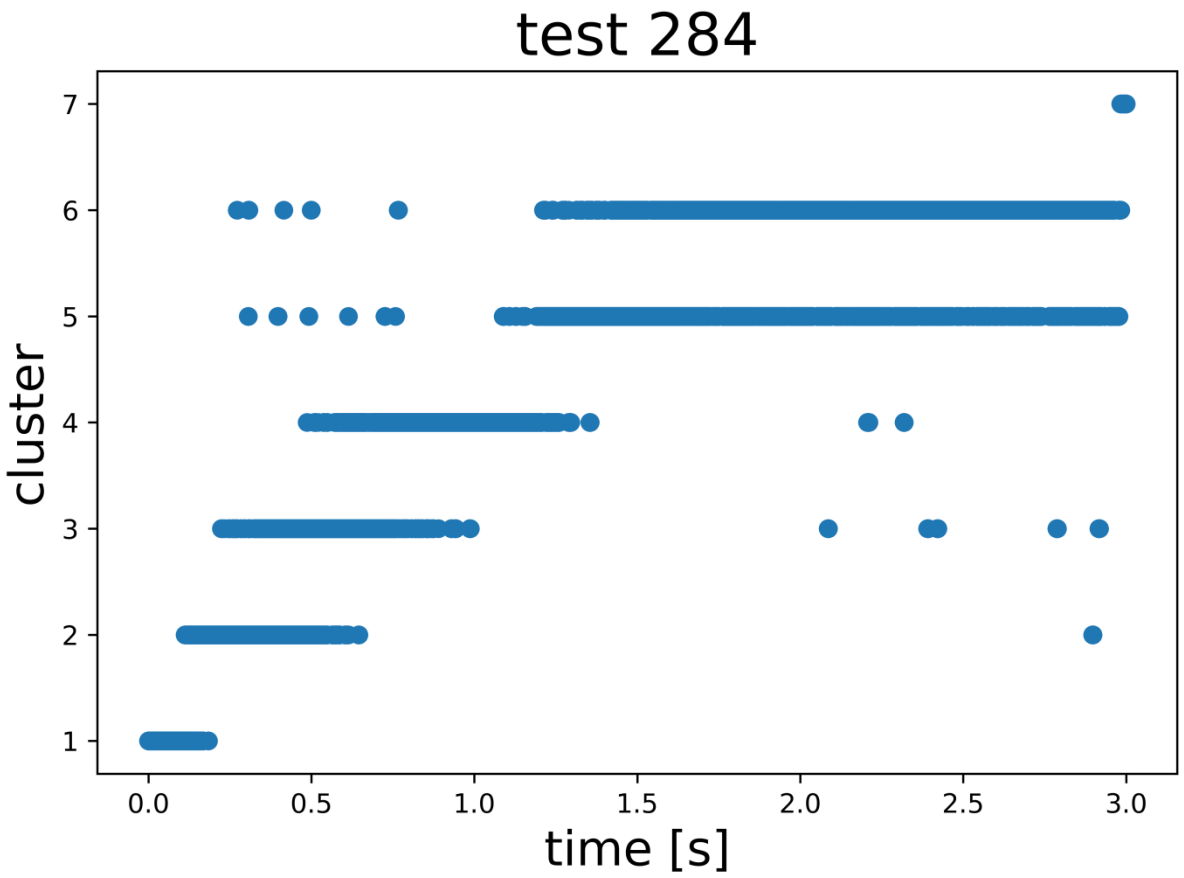



**Fig. 8a:** Spectral clustering with K=7 and affinity matrix from Gaussian kernel with $\sigma = 30000^2$

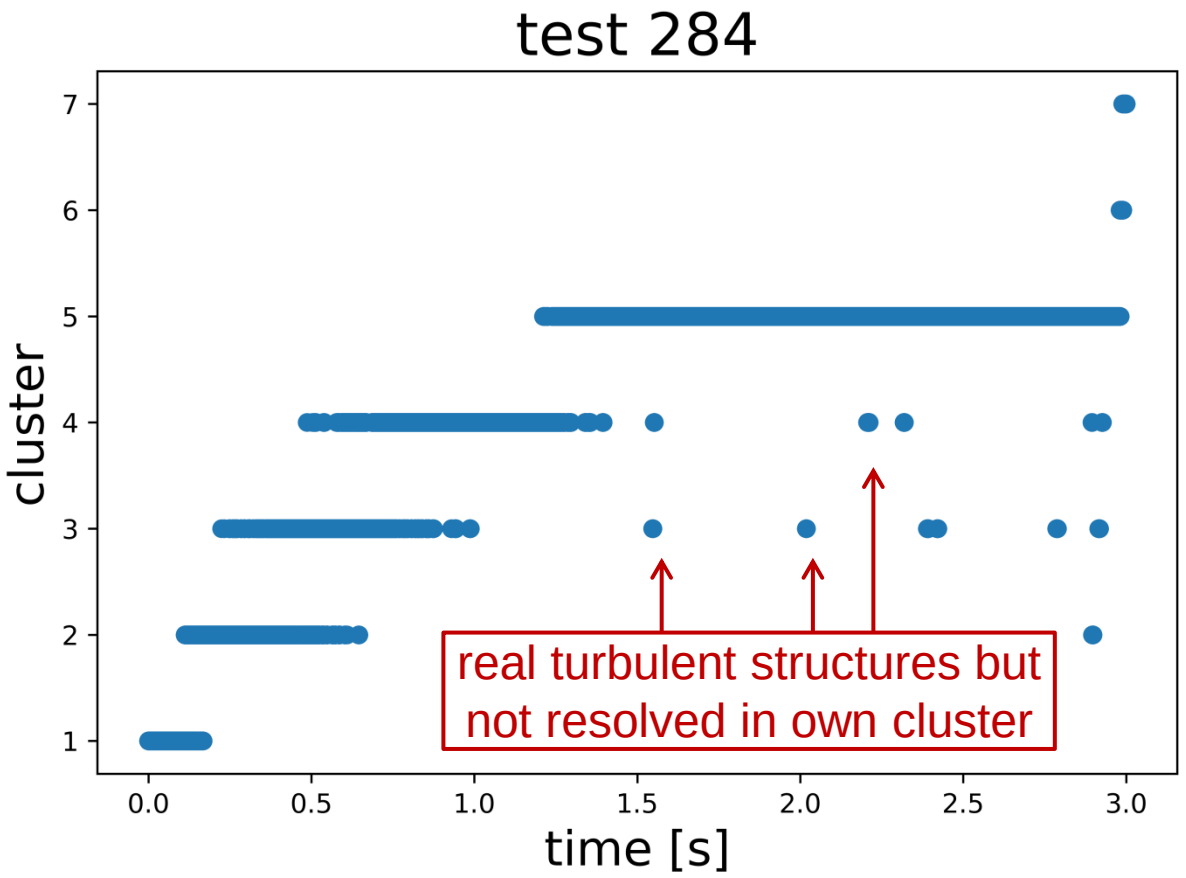



**Fig. 8b:** Analogous clustering with $\sigma = 28000^2$

# Conclusion and outlook

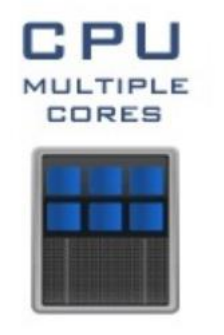
- Clustering of combustion image data with K-means and spectral clustering using HeAT on HPDA-cluster at DLR possible within a reasonable amount of time.
- Analysis of turbulent combustion tests in combustion chamber allows a quantitative test comparison.
- **Future work:** Focus on anomaly detection and more adequate analysis techniques.
- **Further details:**
  - R., Petrarolo and Kobald (2020) *Clustering of Paraffin-Based Hybrid Rocket Fuels Combustion Data*. Experiments in Fluids, 61:4
  - Debus, R., Petrarolo, Kobald and Siggel (2020) *High-performance data analytics of hybrid rocket fuel combustion data using different machine learning approaches*, AIAA SciTech Forum, in press

Thank you for your attention!

# backup slides

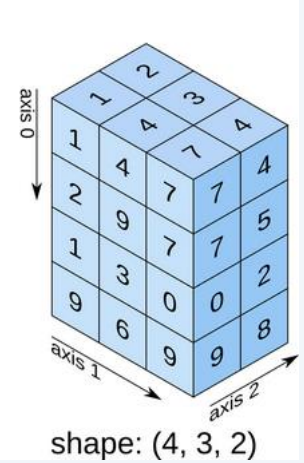
# NumPy

## Runs on




## Data structure

ND-Tensor

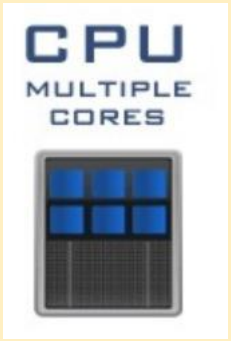



## Operations

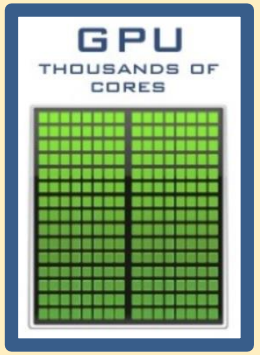
- Elementwise operations
- Slicing
- Matrix operations
- Reduction

# PyTorch

**Runs on**

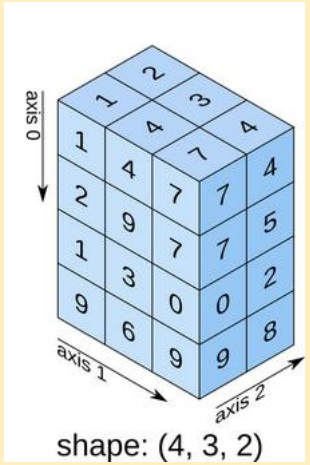



**or**




**Data structure**

ND-Tensor

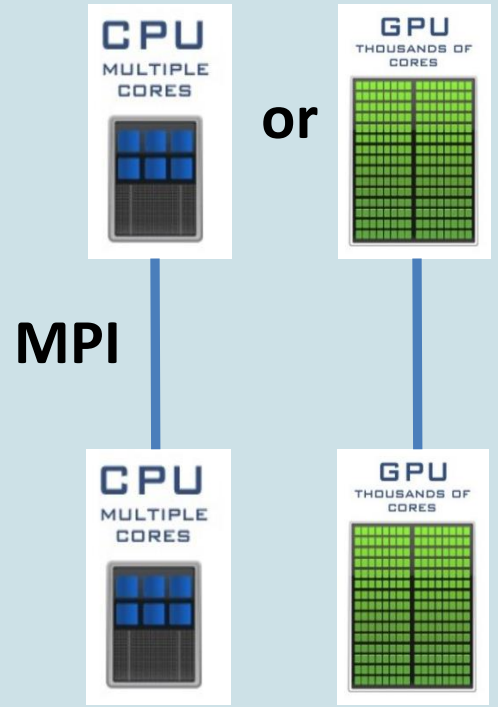



**Operations**

- Elementwise operations
- Slicing
- Matrix operations
- Reduction
- **Automatic differentation**

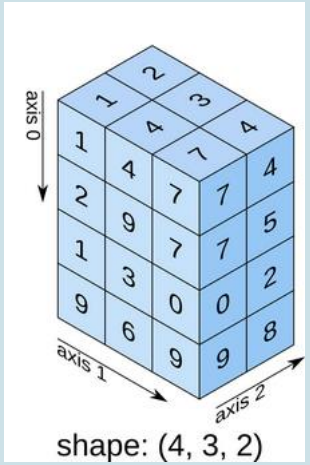
# HeAT

## Runs on




## Data structure

ND-Tensor




## Operations

- Elementwise operations
- Slicing
- Matrix operations
- Reduction
- Automatic differentation